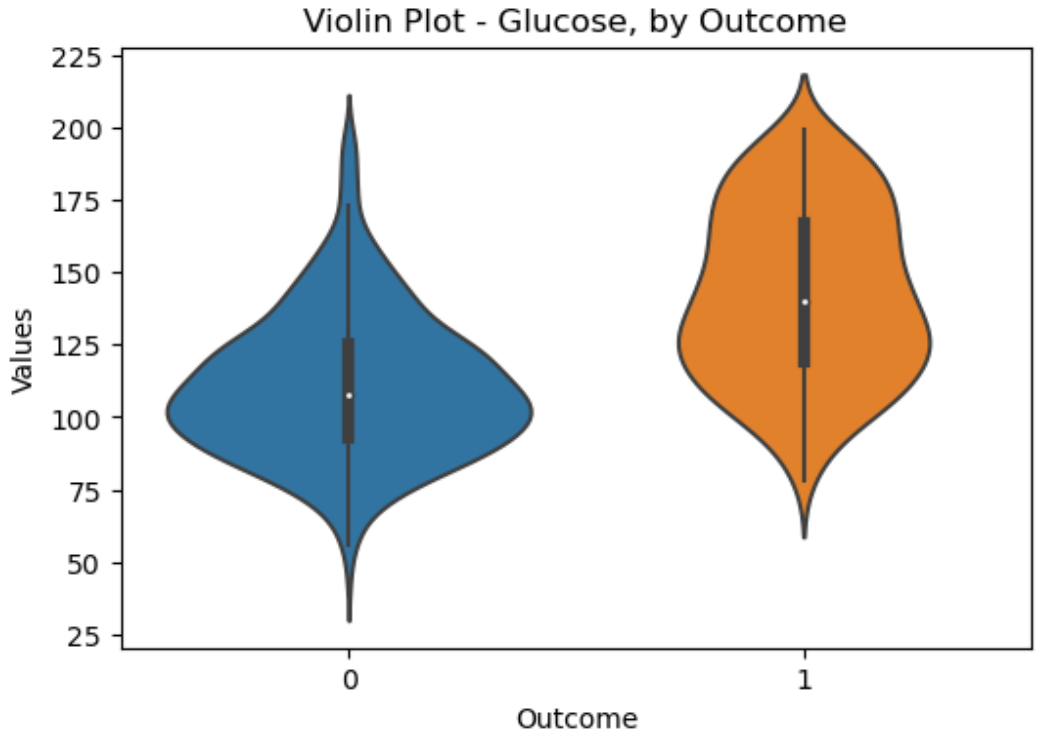
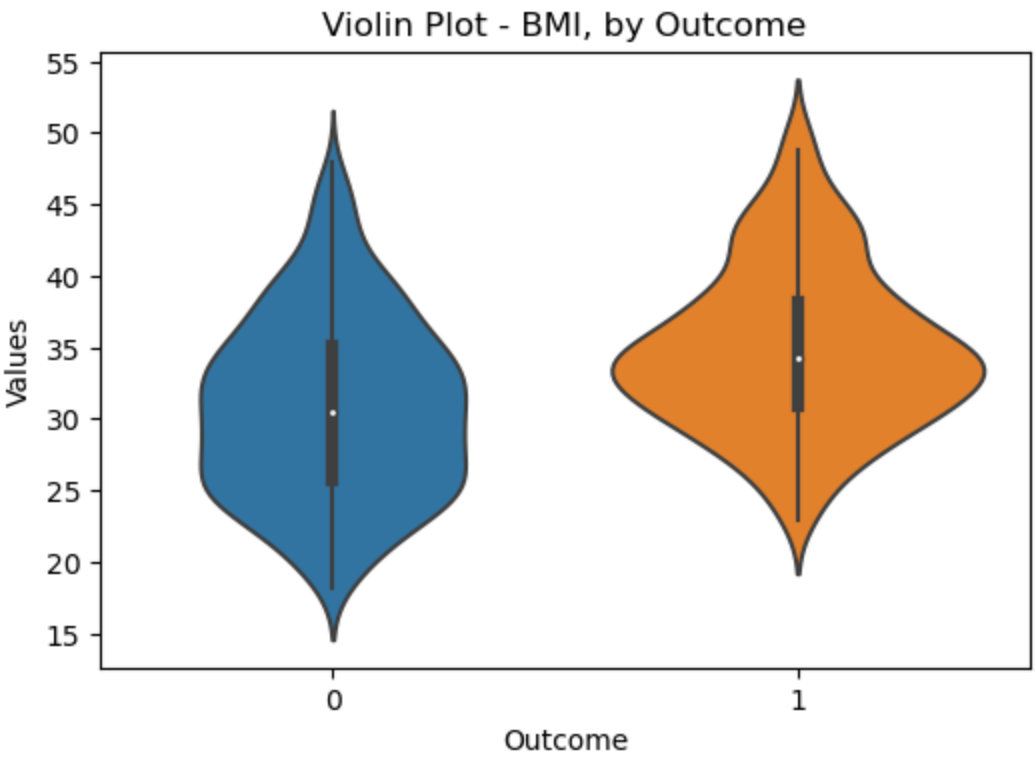
Findings:

The biggest predictors (i.e. what properties most correlate with the presence of diabetes) are Glucose, BMI, Age, number of Pregnancies, and “DiabetesPedigreeFunction” (DPF, i.e. inheritance).

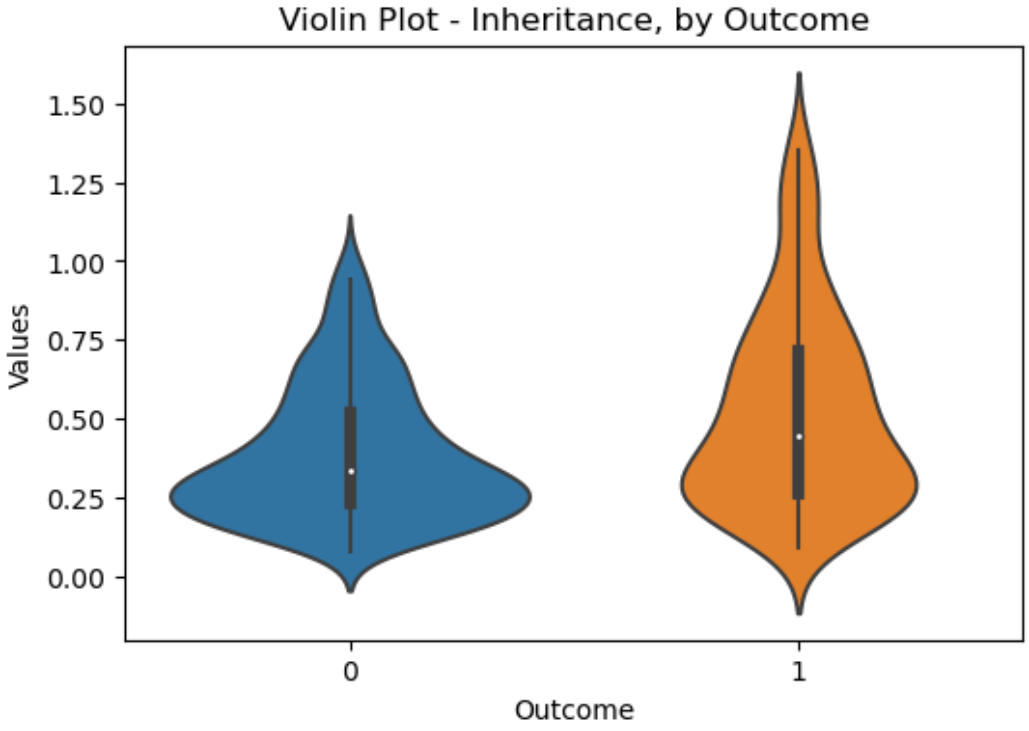
Of these, Glucose is the biggest predictor of diabetes by a large margin, followed by BMI and Age.

Visualizations, with missing values and extreme outliers adjusted to avoid data distortions, are as follows (note that an Outcome of 0 means an individual is non-diabetic, while an Outcome of 1 means an individual is diabetic:




Violin Plot - BMI, by Outcome
55
50
45
40
35
30
25
20
15
Values
0
1
Outcome


Violin Plot - Age, by Outcome
90
80
70
60
50
40
30
20
Values
0
1
Outcome



Stats at a glance:

What is the average age of the individuals in the dataset?

Total avg: 33.24

What is the average glucose level for individuals with diabetes and without diabetes?
Without: 110.68
With: 142.13

Same, for BMI:
Without: 30.78
With: 34.88

Same, for Age:
Without: 31.19
Width, with diabetes: 37.07

Same, for no. of Pregnancies:
Without: 3.09
With: 4.87

Same, for DPF (inheritance):
Without: 0.39
With: 0.52

Other notes:
This dataset has a lot of missing data in it, particularly in the Insulin column. However, missing/null values are indicated by the number '0', which suggests that this was entered in as a default value. For the five columns where this was most suspect (e.g. BMI can never be 0, but no. of Pregnancies can be), I simply

We can get more accurate analysis from our models if we make sure to impute data correctly when it is collected. After all, missing data means that we have no choice but to be less precise

Machine Learning:

To start with, I weighed my options for how to handle my data imbalance and settled on simply adjusting the thresholds for how the models weighed my minority class; in this case, class value 1 = diabetic. While I could have resampled either of my data classes to make them approximately equal in size, I decided that I would not be satisfied with the chance of losing valuable information *or* with overfitting the models to particular data points. As such, I decided to weigh the two Outcome classes differently so that it would be more sensitive to the data found in the smaller class.

I ran three models for this, two of which were individual and one of which was an ensemble model. In all three models, my data was divided into 75% training and 25% test samples, resulting in 192 samples being part of the testing data.

My findings were as follows, with a set seed of 23 for each model:

Logistic Regression:

Confusion Matrix:
[[117 9]
[29 37]]

Classification Report:

	precision	recall	f1-score	support
0	0.80	0.93	0.86	126
1	0.80	0.56	0.66	66
accuracy			0.80	192
macro avg	0.80	0.74	0.76	192
weighted avg	0.80	0.80	0.79	192

AUC Score: 0.882

154 of the test samples were predicted accurately while the remaining 38 were not. My overall accuracy was pretty good, however without a seed I found that I tended to get slightly lower results on average and even saw AUC drop to around 0.82 at times.

Decision Tree:

Confusion Matrix:
[[105 21]
[27 39]]

Classification Report:

	precision	recall	f1-score	support
0	0.80	0.83	0.81	126
1	0.65	0.59	0.62	66
accuracy			0.75	192
macro avg	0.72	0.71	0.72	192
weighted avg	0.75	0.75	0.75	192

AUC Score: 0.712

144 of the test samples were predicted accurately while the remaining 48 were not. My overall accuracy was worse, but without a seed I tended to get consistent yet even worse results (around 0.68 AUC score).

<u>Random Forest</u>: (note: uses a second seed, also set to 23)

Confusion Matrix:
[[122 4]
[32 34]]

Classification Report:

	precision	recall	f1-score	support
0	0.79	0.97	0.87	126
1	0.89	0.52	0.65	66
accuracy			0.81	192
macro avg	0.84	0.74	0.76	192
weighted avg	0.83	0.81	0.80	192

<u>AUC Score</u>: 0.89

This model was my most accurate one, resulting in only 36 incorrect guesses with 100 estimators working in tandem. If I remove one seed then my accuracy and AUC score drop slightly, with AUC averaging around 0.87. This makes it more comparable to the Logistic Regression model without a seed, **but is less variable and therefore a stronger model overall**.